EAST Search History

Ref #	Hits	Search Query	DBs	Default Operator	Plurals	Time Stamp
S91	72	(DLL with (download$4 distribut$3)).ab. (DLL with (download$4 distribut$3)).clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:34
S92	5	S91 not circuit and @rlad<"20031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:04
S93	14	("5210854" "5481713" "6381693" "6463583" "6536038" "6990576" "20030217193").pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:41
S94	0	"winsock.lib" with download$4	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:43
S95	0	("blade server" and library) with (download$4 and upgrad$4)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:44
S96	5	("blade server" and (upgrade firmware library)) same (download$4 broadcast$4)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:49
S97	1	("blade server" and (upgrade firmware) and (download$4 broadcast$3)) and ("management module" same (lib DLL library))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:51
S98	4	("20050071833" "7159105").pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 16:59

S99	15	Rothman.in. and INtel.as. and firmware and routine	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:02
S100	0	S99 and ("management module" UMM)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:01
S101	0	S99 and (DLL)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:01
S102	0	S99 and @pd<"20031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:02
S103	6	(firmware upgrad$4).ab. and @pd<"20031229" and ((server and distribut$4) same library)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:03
S104	1	S103 not circuit and @rlad<"20031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/16 17:04
S105	2	("6266713").URPN.	USPAT	OR	OFF	2007/05/16 17:06
S106	2	("5613151" | "5887190").PN.	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 09:27
S107	16478	intel.as. ("entry point" (pointer near3 table)) and (adverti$5 broadcast$3).clm.	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 09:28
S108	342	S107 and packet and ("entry point" (pointer near3 table)) and (adverti$5 broadcast$3).clm.	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 09:29
S109	10	S108 and ((firmware with (update upgrad$4)) ("blade server"))	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 10:36

S11 0	8	Siemens.as. and firmware and "entry point"	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 10:38
S11 1	0	S110 and (DLL library (dynamic$4 near3 load$4))	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 10:42
S11 2	9	(IAP same firmware)	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 10:42
S11 3	0	S112 and (DLL library "lib" (dynamic$4 near3 load$4))	US-PGPUB; USPAT; USOCR	OR	OFF	2007/05/17 10:42
S11 4	1	(upgrad$4 download$4 firmware) and ((library DLI lib) and ((adverti$4 broadcast$3) same (entry adj point))).clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:28
S11 5	14	(upgrad$4 download$4 firmware) and ((library DLI lib) same (entry adj point)).clm. not S114	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:40
S11 6	0	S115 and intel.as.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:30
S11 7	1	S115 and Rothman.in.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:30
S11 8	13	S115 not S117	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:43
S11 9	2	S118 and @rlad<"20031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:44

EAST Search History

S12 0	11	S118 not S119 and library	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 08:44
S12 1	38	(US-20050021971-$ or US-20040205179-$ or US-20030217193-$ or US-20040255191-$ or US-20040081104-$ or US-20020188713-$ or US-20020010775-$ or US-20050071833-$ or US-20040268109-$ or US-20040243534-$ or US-20030041271-$ or US-20040133622-$ or US-20050138470-$).did. or (US-5724348-$ or US-7017188-$ or US-7007159-$ or US-5664146-$ or US-4570217-$ or US-6301245-$ or US-6918040-$ or US-6311321-$ or US-6401201-$ or US-6381693-$ or US-7127579-$ or US-6463583-$ or US-5481713-$ or US-5210854-$ or US-7159105-$ or US-6266713-$ or US-7039831-$ or US-6247168-$ or US-6266809-$ or US-6317789-$ or US-6868082-$ or US-6957291-$ or US-6983303-$).did. or (US-20020124131-$ or US-20040230785-$).did.	US-PGPUB; USPAT; DERWENT	OR	OFF	2007/05/18 13:52
S12 2	4	S121 and ACPI and ("lib" library DLL)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 14:01
S12 3	72	(EFI ACPI) same (((entry near2 point) pointer) with (table structure))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 14:11
S12 4	2	"6901539".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/18 14:11

S12 5	4	(optimized adj2 librar$3).clm. and (adverti$4 (entry near point)).clm.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:43
S12 6	2	"7159105".pn.	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:44
S12 8	39	((adverti$4 broadcast$4 expos$4 export$3) with ((entry near point) (pointer near3 table))) and (API "application interface") same ((library with dynamic$4 link) DLL)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:49
S12 9	14	S128 and @rlad<"20031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:49
S13 0	25	S128 not S129	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:51
S13 1	18	S130 and (entry with table)	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:52
S13 2	18	S131 and @ad<"200031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:52
S13 3	18	S131 and @ad<="200031229"	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:53

S13 4	1	S133 and ((distribut$3 receiv$4 download$4 retriev$4) near3 (DLL library))	US-PGPUB; USPAT; USOCR; EPO; JPO; DERWENT; IBM_TDB	OR	OFF	2007/05/22 15:54

P⊙RTAL

USPTO

THE ACM DIGITAL LIBRARY

⚡ Feedback Report a problem Satisfaction survey

Terms used **download** **with** **library** and **exporting entry point** and **table pointer**

Found **87,341** of **201,062**

Sort results by	relevance ▾
Display results	expanded form ▾

🔖 Save results to a Binder
❓ Search Tips
☐ Open results in a new window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown   Relevance scale ☐ ◪ ◨ ◼ ■

1 Fast detection of communication patterns in distributed executions   ■
Thomas Kunz, Michiel F. H. Seuren
November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research CASCON '97**
Publisher: IBM Press
Full text available: 📄 pdf(4.21 MB)   Additional Information: full citation, abstract, references, index terms

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

2 GPGPU: general purpose computation on graphics hardware   ■
David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff Woolley, Aaron Lefohn
August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**
Publisher: ACM Press
Full text available: 📄 pdf(63.03 MB)   Additional Information: full citation, abstract, citings

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

3 Exokernel: an operating system architecture for application-level resource management   ■
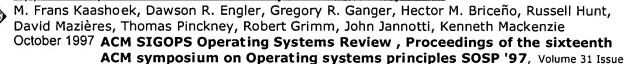D. R. Engler, M. F. Kaashoek, J. O'Toole
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5
Publisher: ACM Press
Full text available: 📄 pdf(2.16 MB)   Additional Information: full citation, references, citings, index terms

4 Application performance and flexibility on exokernel systems

M. Frans Kaashoek, Dawson R. Engler, Gregory R. Ganger, Hector M. Briceño, Russell Hunt, David Mazières, Thomas Pinckney, Robert Grimm, John Jannotti, Kenneth Mackenzie

October 1997 **ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles SOSP '97**, Volume 31 Issue 5

Publisher: ACM Press

Full text available: pdf(2.39 MB)   Additional Information: full citation, references, citings, index terms

5 Dynamic software updating

Michael Hicks, Scott Nettles

November 2005 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 27 Issue 6

Publisher: ACM Press

Full text available: pdf(622.69 KB)   Additional Information: full citation, abstract, references, index terms

Many important applications must run continuously and without interruption, and yet also must be changed to fix bugs or upgrade functionality. No prior general-purpose methodology for dynamic updating achieves a practical balance between flexibility, robustness, low overhead, ease of use, and low cost.We present an approach for C-like languages that provides type-safe dynamic updating of native code in an extremely flexible manner---code, data, and types may be updated, at programmer-determined ...

Keywords: Dynamic software updating, typed assembly language

6 The KaffeOS Java runtime system

Godmar Back, Wilson C. Hsieh

July 2005 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 27 Issue 4

Publisher: ACM Press

Full text available: pdf(704.30 KB)   Additional Information: full citation, abstract, references, citings, index terms, review

Single-language runtime systems, in the form of Java virtual machines, are widely deployed platforms for executing untrusted mobile code. These runtimes provide some of the features that operating systems provide: interapplication memory protection and basic system services. They do not, however, provide the ability to isolate applications from each other. Neither do they provide the ability to limit the resource consumption of applications. Consequently, the performance of current systems degra ...

Keywords: Robustness, garbage collection, isolation, language runtimes, resource management, termination, virtual machines

7 SPiKE: engineering malware analysis tools using unobtrusive binary-instrumentation

Amit Vasudevan, Ramesh Yerraballi

January 2006 **Proceedings of the 29th Australasian Computer Science Conference - Volume 48 ACSC '06**

Publisher: Australian Computer Society, Inc.

Full text available: pdf(832.66 KB)   Additional Information: full citation, abstract, references, index terms

Malware -- a generic term that encompasses viruses, trojans, spywares and other intrusive code -- is widespread today. Malware analysis is a multi-step process providing insight into malware structure and functionality, facilitating the development of an

antidote. Behavior monitoring, an important step in the analysis process, is used to observe malware interaction with respect to the system and is achieved by employing dynamic coarse-grained binary-instrumentation on the target system. However, ...

Keywords: instrumentation, malware, security

8   Fast and flexible application-level networking on exokernel systems

Gregory R. Ganger, Dawson R. Engler, M. Frans Kaashoek, Hector M. Briceño, Russell Hunt, Thomas Pinckney
February 2002 **ACM Transactions on Computer Systems (TOCS)**, Volume 20 Issue 1
Publisher: ACM Press

Full text available: 📄 pdf(500.67 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Application-level networking is a promising software organization for improving performance and functionality for important network services. The Xok/ExOS exokernel system includes application-level support for standard network services, while at the same time allowing application writers to specialize networking services. This paper describes how Xok/ExOS's kernel mechanisms and library operating system organization achieve this flexibility, and retrospectively shares our experiences an ...

Keywords: Extensible systems, OS structure, fast servers, network services

9   Wide-area cooperative storage with CFS

Frank Dabek, M. Frans Kaashoek, David Karger, Robert Morris, Ion Stoica
October 2001 **ACM SIGOPS Operating Systems Review , Proceedings of the eighteenth ACM symposium on Operating systems principles SOSP '01**, Volume 35 Issue 5
Publisher: ACM Press

Full text available: 📄 pdf(1.25 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

The Cooperative File System (CFS) is a new peer-to-peer read-only storage system that provides provable guarantees for the efficiency, robustness, and load-balance of file storage and retrieval. CFS does this with a completely decentralized architecture that can scale to large systems. CFS servers provide a distributed hash table (DHash) for block storage. CFS clients interpret DHash blocks as a file system. DHash distributes and caches blocks at a fine granularity to achieve load balance, uses ...

10   High dynamic range imaging

Paul Debevec, Erik Reinhard, Greg Ward, Sumanta Pattanaik
August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**
Publisher: ACM Press
Full text available: 📄 pdf(20.22 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>

Current display devices can display only a limited range of contrast and colors, which is one of the main reasons that most image acquisition, processing, and display techniques use no more than eight bits per color channel. This course outlines recent advances in high-dynamic-range imaging, from capture to display, that remove this restriction, thereby enabling images to represent the color gamut and dynamic range of the original scene rather than the limited subspace imposed by current monitor ...

11   Programming languages for mobile code

Tommy Thorn
September 1997 **ACM Computing Surveys (CSUR)**, Volume 29 Issue 3

Publisher: ACM Press

Full text available: pdf(393.65 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

Sun's announcement of the programming language Java more that anything popularized the notion of mobile code, that is, programs traveling on a heterogeneous network and automatically executing upon arrival at the destination. We describe several classes of mobile code and extract their common characteristics, where security proves to be one of the major concerns. With these characteristics as reference points, we examine six representative languages proposed for mobile code. The conclusion ...

Keywords: Java, Limbo, Objective Caml, Obliq, Safe-Tcl, distribution, formal methods, mobile code, network programming, object orientation, portability, safety, security, telescript

12 <u>EROS: a fast capability system</u>
Jonathan S. Shapiro, Jonathan M. Smith, David J. Farber
December 1999 **ACM SIGOPS Operating Systems Review , Proceedings of the seventeenth ACM symposium on Operating systems principles SOSP '99**, Volume 33 Issue 5
Publisher: ACM Press

Full text available: pdf(1.83 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

EROS is a capability-based operating system for commodity processors which uses a single level storage model. The single level store's persistence is transparent to applications. The performance consequences of support for transparent persistence and capability-based architectures are generally believed to be negative. Surprisingly, the basic operations of EROS (such as IPC) are generally comparable in cost to similar operations in conventional systems. This is demonstrated with a set of microbe ...

13 <u>An open-source CVE for programming education: a case study: An open-source CVE for programming education: a case study</u>
Andrew M. Phelps, Christopher A. Egert, Kevin J. Bierre, David M. Parks
July 2005 **ACM SIGGRAPH 2005 Courses SIGGRAPH '05**
Publisher: ACM Press
Full text available: pdf(7.92 MB)   Additional Information: <u>full citation</u>, <u>references</u>

14 <u>Rover: a toolkit for mobile information access</u>
A. D. Joseph, A. F. de Lespinasse, J. A. Tauber, D. K. Gifford, M. F. Kaashoek
December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5
Publisher: ACM Press
Full text available: pdf(2.18 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

15 <u>Distributed operating systems</u>
Andrew S. Tanenbaum, Robbert Van Renesse
December 1985 **ACM Computing Surveys (CSUR)**, Volume 17 Issue 4
Publisher: ACM Press

Full text available: pdf(5.49 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

Distributed operating systems have many aspects in common with centralized ones, but they also differ in certain ways. This paper is intended as an introduction to distributed operating systems, and especially to current university research about them. After a discussion of what constitutes a distributed operating system and how it is distinguished from a computer network, various key design issues are discussed. Then several examples of current research projects are examined in some detail ...

16 A principled approach to operating system construction in Haskell

Thomas Hallgren, Mark P. Jones, Rebekah Leslie, Andrew Tolmach
September 2005 **ACM SIGPLAN Notices , Proceedings of the tenth ACM SIGPLAN international conference on Functional programming ICFP '05**, Volume 40 Issue 9

Publisher: ACM Press

Full text available: pdf(154.82 KB)   Additional Information: full citation, abstract, references, index terms

We describe a monadic interface to low-level hardware features that is a suitable basis for building operating systems in Haskell. The interface includes primitives for controlling memory management hardware, user-mode process execution, and low-level device I/O. The interface enforces memory safety in nearly all circumstances. Its behavior is specified in part by formal assertions written in a programming logic called P-Logic. The interface has been implemented on bare IA32 hardware using the G ...

Keywords: Haskell, hardware interface, monads, operating systems, programming logic, verification

17 The elements of nature: interactive and realistic techniques

Oliver Deusen, David S. Ebert, Ron Fedkiw, F. Kenton Musgrave, Przemyslaw Prusinkiewicz, Doug Roble, Jos Stam, Jerry Tessendorf
August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: pdf(17.65 MB)   Additional Information: full citation, abstract

This updated course on simulating natural phenomena will cover the latest research and production techniques for simulating most of the elements of nature. The presenters will provide movie production, interactive simulation, and research perspectives on the difficult task of photorealistic modeling, rendering, and animation of natural phenomena. The course offers a nice balance of the latest interactive graphics hardware-based simulation techniques and the latest physics-based simulation techni ...

18 Developing mobile 3D applications with OpenGL ES and M3G: Developing mobile 3D applications with OpenGL ES and M3G

Kari Pulli, Jani Vaarala, Ville Miettinen, Tomi Aarnio, Mark Callow
July 2005 **ACM SIGGRAPH 2005 Courses SIGGRAPH '05**

Publisher: ACM Press

Full text available: pdf(9.22 MB)   Additional Information: full citation

19 Versatility and Unix semantics in namespace unification

Charles P. Wright, Jay Dave, Puja Gupta, Harikesavan Krishnan, David P. Quigley, Erez Zadok, Mohammad Nayyer Zubair
February 2006 **ACM Transactions on Storage (TOS)**, Volume 2 Issue 1

Publisher: ACM Press

Full text available: pdf(317.82 KB)   Additional Information: full citation, abstract, references, citings, index terms

Administrators often prefer to keep related sets of files in different locations or media, as it is easier to maintain them separately. Users, however, prefer to see all files in one location for convenience. One solution that accommodates both needs is virtual namespace unification---providing a merged view of several directories without physically merging them. For example, namespace unification can merge the contents of several CD-ROM images without unpacking them, merge binary directories fr ...

Keywords: Namespace management, directory merging, snapshotting, stackable file systems, unification

20 Crowd and group animation

Daniel Thalmann, Christophe Hery, Seth Lippman, Hiromi Ono, Stephen Regelous, Douglas Sutton

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: pdf(20.19 MB)   Additional Information: full citation, abstract

A continuous challenge for special effects in movies is the production of realistic virtual crowds, in terms of rendering and behavior. This course will present state-of-the-art techniques and methods. The course will explain in details the different approaches to create virtual crowds: particle systems with flocking techniques using attraction and repulsion forces, copy and pasting techniques, agent-based methods. The architecture of software tools will be presented including the MASSIVE softwa ...

Results 1 - 20 of 200   Result page: **1**   2   3   4   5   6   7   8   9   10   next

PORTAL

USPTO

Search: ⦿ The ACM Digital Library   ○ The Guide

(download with library) and (entry point") and "export table" a   **SEARCH**

THE ACM DIGITAL LIBRARY

○< Feedback Report a problem Satisfaction
survey

Terms used **download with library** and **entry point** and **export
table** and **optimized library**

Found **128,092** of **201,062**

| Sort results by | relevance ▾ |
| Display results | expanded form ▾ |

◆ Save results to a Binder
? Search Tips
☐ Open results in a new
window

Try an Advanced Search
Try this search in The ACM Guide

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10   next
Best 200 shown

Relevance scale ☐ ▭ ▬ ▬ ■

1   SPiKE: engineering malware analysis tools using unobtrusive binary-instrumentation   ■
   Amit Vasudevan, Ramesh Yerraballi
   January 2006 **Proceedings of the 29th Australasian Computer Science Conference -
   Volume 48 ACSC '06**
   Publisher: Australian Computer Society, Inc.
   Full text available: 🗎 pdf(832.66 KB)   Additional Information: full citation, abstract, references, index terms

   Malware -- a generic term that encompasses viruses, trojans, spywares and other
   intrusive code -- is widespread today. Malware analysis is a multi-step process providing
   insight into malware structure and functionality, facilitating the development of an
   antidote. Behavior monitoring, an important step in the analysis process, is used to
   observe malware interaction with respect to the system and is achieved by employing
   dynamic coarse-grained binary-instrumentation on the target system. However, ...

   Keywords: instrumentation, malware, security

2   GALAHAD, a library of thread-safe Fortran 90 packages for large-scale nonlinear   ■
◈ optimization
   Nicholas I. M. Gould, Dominique Orban, Philippe L. Toint
   December 2003 **ACM Transactions on Mathematical Software (TOMS)**, Volume 29 Issue 4
   Publisher: ACM Press
   Full text available: 🗎 pdf(146.51 KB)   Additional Information: full citation, abstract, references, index terms

   We describe the design of version 1.0 of GALAHAD, a library of Fortran 90 packages for
   large-scale nonlinear optimization. The library particularly addresses quadratic
   programming problems, containing both interior point and active set algorithms, as well as
   tools for preprocessing problems prior to solution. It also contains an updated version of
   the venerable nonlinear programming package, LANCELOT.

   Keywords: Fortran 90, GALAHAD, LANCELOT, large-scale nonlinear optimization, large-
   scale quadratic programming

3   GPGPU: general purpose computation on graphics hardware   ■
◈ David Luebke, Mark Harris, Jens Krüger, Tim Purcell, Naga Govindaraju, Ian Buck, Cliff
   Woolley, Aaron Lefohn

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: 📄 pdf(63.03 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>citings</u>

The graphics processor (GPU) on today's commodity video cards has evolved into an extremely powerful and flexible processor. The latest graphics architectures provide tremendous memory bandwidth and computational horsepower, with fully programmable vertex and pixel processing units that support vector operations up to full IEEE floating point precision. High level languages have emerged for graphics hardware, making this computational power accessible. Architecturally, GPUs are highly parallel s ...

4 Fast detection of communication patterns in distributed executions ■

Thomas Kunz, Michiel F. H. Seuren

November 1997 **Proceedings of the 1997 conference of the Centre for Advanced Studies on Collaborative research CASCON '97**

Publisher: IBM Press

Full text available: 📄 pdf(4.21 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Understanding distributed applications is a tedious and difficult task. Visualizations based on process-time diagrams are often used to obtain a better understanding of the execution of the application. The visualization tool we use is Poet, an event tracer developed at the University of Waterloo. However, these diagrams are often very complex and do not provide the user with the desired overview of the application. In our experience, such tools display repeated occurrences of non-trivial commun ...

5 Extracting library-based object-oriented applications ■

Peter F. Sweeney, Frank Tip

November 2000 **ACM SIGSOFT Software Engineering Notes , Proceedings of the 8th ACM SIGSOFT international symposium on Foundations of software engineering: twenty-first century applications SIGSOFT '00/FSE-8,** Volume 25 Issue 6

Publisher: ACM Press

Full text available: 📄 pdf(1.06 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

In an increasingly popular model of software distribution, software is developed in one computing environment and deployed in other environments by transfer over the internet. Extraction tools perform a static whole-program analysis to determine unused functionality in applications in order to reduce the time required to download applications. We have identified a number of scenarios where extraction tools require information beyond what can be inferred through static analysis: software distr ...

6 High dynamic range imaging ■

Paul Debevec, Erik Reinhard, Greg Ward, Sumanta Pattanaik

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: 📄 pdf(20.22 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>

Current display devices can display only a limited range of contrast and colors, which is one of the main reasons that most image acquisition, processing, and display techniques use no more than eight bits per color channel. This course outlines recent advances in high-dynamic-range imaging, from capture to display, that remove this restriction, thereby enabling images to represent the color gamut and dynamic range of the original scene rather than the limited subspace imposed by current monitor ...

7 Practical extraction techniques for Java ■

Frank Tip, Peter F. Sweeney, Chris Laffra, Aldo Eisma, David Streeter

November 2002 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 24 Issue 6
Publisher: ACM Press

Full text available: pdf(1.01 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

Reducing application size is important for software that is distributed via the internet, in order to keep download times manageable, and in the domain of embedded systems, where applications are often stored in (Read-Only or Flash) memory. This paper explores extraction techniques such as the removal of unreachable methods and redundant fields, inlining of method calls, and transformation of the class hierarchy for reducing application size. We implemented a number of extraction techniques in < ...

Keywords: Application extraction, call graph construction, class hierarchy transformation, packaging, whole-program analysis

8 Stack allocation and synchronization optimizations for Java using escape analysis ▪

Jong-Deok Choi, Manish Gupta, Mauricio J. Serrano, Vugranam C. Sreedhar, Samuel P. Midkiff
November 2003 **ACM Transactions on Programming Languages and Systems (TOPLAS)**, Volume 25 Issue 6
Publisher: ACM Press

Full text available: pdf(632.85 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>, <u>review</u>

This article presents an *escape analysis* framework for Java to determine (1) if an object is not reachable after its method of creation returns, allowing the object to be allocated on the stack, and (2) if an object is reachable only from a single thread during its lifetime, allowing unnecessary synchronization operations on that object to be removed. We introduce a new program abstraction for escape analysis, the *connection graph*, that is used to establish reachability relationshi ...

Keywords: Connection graphs, escape analysis, points-to graph

9 Link and channel measurement: A simple mechanism for capturing and replaying wireless channels ▪

Glenn Judd, Peter Steenkiste
August 2005 **Proceeding of the 2005 ACM SIGCOMM workshop on Experimental approaches to wireless network design and analysis E-WIND '05**
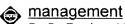**Publisher:** ACM Press

Full text available: pdf(6.06 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

Physical layer wireless network emulation has the potential to be a powerful experimental tool. An important challenge in physical emulation, and traditional simulation, is to accurately model the wireless channel. In this paper we examine the possibility of using on-card signal strength measurements to capture wireless channel traces. A key advantage of this approach is the simplicity and ubiquity with which these measurements can be obtained since virtually all wireless devices provide the req ...

Keywords: channel capture, emulation, wireless

10 Exokernel: an operating system architecture for application-level resource management ▪

D. R. Engler, M. F. Kaashoek, J. O'Toole

December 1995 **ACM SIGOPS Operating Systems Review , Proceedings of the fifteenth ACM symposium on Operating systems principles SOSP '95**, Volume 29 Issue 5

Publisher: ACM Press

Full text available: pdf(2.16 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

---

11   Practical experience with an application extractor for Java

Frank Tip, Chris Laffra, Peter F. Sweeney, David Streeter

October 1999 **ACM SIGPLAN Notices , Proceedings of the 14th ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications OOPSLA '99**, Volume 34 Issue 10

Publisher: ACM Press

Full text available: pdf(2.31 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Java programs are routinely transmitted over low-bandwidth network connections as compressed class file archives (i.e., zip files and jar files). Since archive size is directly proportional to download time, it is desirable for applications to be as small as possible. This paper is concerned with the use of program transformations such as removal of dead methods and fields, inlining of method calls, and simplification of the class hierarchy for reducing application size. Such "extract ...

---

12   Application performance and flexibility on exokernel systems

M. Frans Kaashoek, Dawson R. Engler, Gregory R. Ganger, Hector M. Briceño, Russell Hunt, David Mazières, Thomas Pinckney, Robert Grimm, John Jannotti, Kenneth Mackenzie

October 1997 **ACM SIGOPS Operating Systems Review , Proceedings of the sixteenth ACM symposium on Operating systems principles SOSP '97**, Volume 31 Issue 5

Publisher: ACM Press

Full text available: pdf(2.39 MB)   Additional Information: <u>full citation</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

---

13   Real-time shading

Marc Olano, Kurt Akeley, John C. Hart, Wolfgang Heidrich, Michael McCool, Jason L. Mitchell, Randi Rost

August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**

Publisher: ACM Press

Full text available: pdf(7.39 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>

Real-time procedural shading was once seen as a distant dream. When the first version of this course was offered four years ago, real-time shading was possible, but only with one-of-a-kind hardware or by combining the effects of tens to hundreds of rendering passes. Today, almost every new computer comes with graphics hardware capable of interactively executing shaders of thousands to tens of thousands of instructions. This course has been redesigned to address today's real-time shading capabili ...

---

14   Fast and flexible application-level networking on exokernel systems

Gregory R. Ganger, Dawson R. Engler, M. Frans Kaashoek, Hector M. Briceño, Russell Hunt, Thomas Pinckney

February 2002 **ACM Transactions on Computer Systems (TOCS)**, Volume 20 Issue 1

Publisher: ACM Press

Full text available: pdf(500.67 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

Application-level networking is a promising software organization for improving

performance and functionality for important network services. The Xok/ExOS exokernel system includes application-level support for standard network services, while at the same time allowing application writers to specialize networking services. This paper describes how Xok/ExOS's kernel mechanisms and library operating system organization achieve this flexibility, and retrospectively shares our experiences an ...

Keywords: Extensible systems, OS structure, fast servers, network services

15 NEOS and Condor: solving optimization problems over the Internet

Michael C. Ferris, Michael P. Mesnier, Jorge J. Moré
March 2000 **ACM Transactions on Mathematical Software (TOMS)**, Volume 26 Issue 1
Publisher: ACM Press

Full text available: pdf(125.03 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>citings</u>, <u>index terms</u>

We discuss the use of Condor, a distributed resource management system, as a provider of computational resources for NEOS, an environment for solving optimization problems over the Internet. We also describe how problems are submitted and processed by NEOS, and then scheduled and solved by Condor on available (idle) workstations

Keywords: automatic differentiation, complementarity problems, computational servers, network computing, resource management

16 Balancing performance and flexibility with hardware support for network architectures

Ilija Hadžić, Jonathan M. Smith
November 2003 **ACM Transactions on Computer Systems (TOCS)**, Volume 21 Issue 4
Publisher: ACM Press

Full text available: pdf(719.03 KB)   Additional Information: <u>full citation</u>, <u>abstract</u>, <u>references</u>, <u>index terms</u>

The goals of performance and flexibility are often at odds in the design of network systems. The tension is common enough to justify an architectural solution, rather than a set of context-specific solutions. The Programmable Protocol Processing Pipeline (P4) design uses programmable hardware to selectively accelerate protocol processing functions. A set of field-programmable gate arrays (FPGAs) and an associated library of network processing modules implemented in hardware are augmented with so ...

Keywords: FPGA, P4, computer networking, flexibility, hardware, performance, programmable logic devices, programmable networks, protocol processing

17 The elements of nature: interactive and realistic techniques

Oliver Deusen, David S. Ebert, Ron Fedkiw, F. Kenton Musgrave, Przemyslaw Prusinkiewicz, Doug Roble, Jos Stam, Jerry Tessendorf
August 2004 **ACM SIGGRAPH 2004 Course Notes SIGGRAPH '04**
Publisher: ACM Press

Full text available: pdf(17.65 MB)   Additional Information: <u>full citation</u>, <u>abstract</u>

This updated course on simulating natural phenomena will cover the latest research and production techniques for simulating most of the elements of nature. The presenters will provide movie production, interactive simulation, and research perspectives on the difficult task of photorealistic modeling, rendering, and animation of natural phenomena. The course offers a nice balance of the latest interactive graphics hardware-based simulation techniques and the latest physics-based simulation techni ...

18

Memory overflow protection for embedded systems using run-time checks, reuse, and compression

Surupa Biswas, Thomas Carley, Matthew Simpson, Bhuvan Middha, Rajeev Barua
November 2006 **ACM Transactions on Embedded Computing Systems (TECS)**, Volume 5 Issue 4
Publisher: ACM Press
Full text available: pdf(579.85 KB)   Additional Information: full citation, abstract, references, index terms

Embedded systems usually lack virtual memory and are vulnerable to memory overflow since they lack a mechanism to detect overflow or use swap space thereafter. We present a method to detect memory overflows using compiler-inserted software run-time checks. Its overheads in run-time and energy are 1.35 and 1.12%, respectively. Detection of overflow allows system-specific remedial action. We also present techniques to grow the stack or heap segment after they overflow, into previously unuti ...

Keywords: Out-of-memory errors, data compression, heap overflow, reliability, reuse, run-time checks, stack overflow

19 Shared memory programming for large scale machines

Christopher Barton, CĆlin Casçaval, George Almási, Yili Zheng, Montse Farreras, Siddhartha Chatterje, José Nelson Amaral
June 2006 **ACM SIGPLAN Notices , Proceedings of the 2006 ACM SIGPLAN conference on Programming language design and implementation PLDI '06**, Volume 41 Issue 6
Publisher: ACM Press
Full text available: pdf(245.02 KB)   Additional Information: full citation, abstract, references, index terms

This paper describes the design and implementation of a scalable run-time system and an optimizing compiler for Unified Parallel C (UPC). An experimental evaluation on BlueGene/L®, a distributed-memory machine, demonstrates that the combination of the compiler with the runtime system produces programs with performance comparable to that of efficient MPI programs and good performance scalability up to hundreds of thousands of processors.Our runtime system design solves the problem of maintai ...

Keywords: BlueGene, PGAS programming model, UPC

20 MicroLib: A Case for the Quantitative Comparison of Micro-Architecture Mechanisms

Daniel Gracia Perez, Gilles Mouchard, Olivier Temam
December 2004 **Proceedings of the 37th annual IEEE/ACM International Symposium on Microarchitecture MICRO 37**
Publisher: IEEE Computer Society
Full text available: pdf(917.33 KB)   Additional Information: full citation, abstract, citings

While most research papers on computer architectures include some performance measurements, these performance numbers tend to be distrusted. Up to the point that, after so many research articles on data cache architectures, for instance, few researchers have a clear view of what are the best data cache mechanisms. To illustrate the usefulness of a fair quantitative comparison, we have picked a target architecture component for which lots of optimizations have been proposed (data caches), and we ...

Results 1 - 20 of 200   Result page: **1** 2 3 4 5 6 7 8 9 10 next

Useful downloads: Adobe Acrobat   QuickTime   Windows Media Player   Real Player